

# w3af

## better than a regular script kiddie

Mark Keegan

Security Consultant

# Talk Objectives

- What is w3af
- High level introduction to the tool



“Why do web app security testers need yet another tool”

# Who am I

- Security Consultant
- Focusing on WebApp Sec
- Performed many web application pentests for both local and international companies

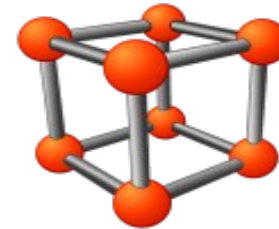


# Web App Sec Challenges

- Custom applications
- Large complex applications with many technologies
- Typical tests are performed at the end of the SDLC
- Time to perform the assessment is limited
- Statement of Works define the scope and limitations
- A malicious hacker does not have the same constraints



# Solution



Process + Experience + Framework

Create a framework where everyone can contribute with his knowledge.

The framework must allow for easy integration of new features

# Problems with current tools

Commercial tools

VS

Open Source



# Enter w3af

- **W**eb **A**pplication **A**ttack and **A**udit **F**ramework
- Discovery / Audit and Exploit tool
- Open Source - General Public License v2
- Andres Riancho
- Python - design patterns
- w3af beta5 Oct 2007 - code named YEN with some very cool new features:)
- 110 plugins (and growing)



# The core of w3af

- Provides developers with a reusable library
- Coordinates requests to the remote server
- Communicates with other plugins through the Knowledge Base

# w3af Phases

The framework is divided into 3 key phases. These are at the root of any penetration test:

- **Discovery**

Find new URLs, forms and other injection points, they also find important information about the remote web application users, web server and programming framework

- **Audit**

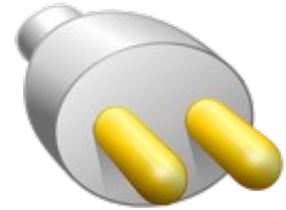
Takes injection points and finds vulnerabilities

- **Attack**

Abuse found vulnerabilities and return something useful

# Other Plugins

- **grep**  
finds vulnerabilities in page content
- **output**  
communicates results to the user
- **mangle**  
modify requests and responses based on regular expressions
- **evasion**  
evades simple intrusion detection rules
- **bruteforce**  
brute forces forms and basic authentication



# Grep and Bruteforce Demo

Lets test the strength of a forms authenticated page...

- Scenario:
  - we have located a page containing a login form. We want to test the strength of this login.
  - do not know a valid username or pwd.

# Other Discovery Plugins

- Discovery plugins main responsibility is to find new URLs, forms, and other injection points.
- WebSpider
- ServerStatus
- Halberd – finds HTTP load balancers
- phpEggs – fingerprints using known php eggs
- Pykto – python version of Nikto
- archiveDotOrg – Searches archive.org to find new URLs



# urlFuzzer

This plugin will try to find new URL's based on the URL for example:

`http://a/a.html`

The plugin will request:

- `http://a/a.html.tgz`
- `http://a/a.tgz`
- `http://a/a.zip`
- ... etc

If the response is not a 404 error, then we have found a new URL.

# wordNet

This plugin finds new URL's using wordnet.

let's suppose that the input for this plugin is:

- `http://a/index.asp?color=blue`

The plugin will search the wordnet database for words that are related with "blue", and return for example: "black" and "white". So the plugin requests this two URL's:

- `http://a/index.asp?color=black`
- `http://a/index.asp?color=white`

# SpiderMan

- This plugin is a local proxy that can be used to give the framework knowledge about the web application when it has a lot of client side code like Flash or Java applets. Whenever a w3af needs to test an application with flash or javascript, the user should enable this plugin and use a web browser to navigate the site using spiderMan proxy.
- The proxy will extract information from the user navigation and generate the necessary injection points for the audit plugins.
- Saves the cookies that are sent by the web application, in order to be able to use them in other plugins.

# Audit: a closer look

- Tests injection points found in the discovery phase for vulnerabilities.

```
w3af/plugins>>> list audit
xpath          This plugin tests for XPATH injection bugs.
LDAPi          This plugin tests for LDAP injection bugs.
sslCertificate This plugin checks the SSL certificate validity( if https is being used ).
osCommanding  This plugin tests for OS Commanding bugs.
responseSplitting This plugin tests for response splitting bugs.
mxInjection    This plugin tests for MX injection bugs.
blindSqli      This plugin tests for blind SQL injection bugs.
formatString   This plugin tests for format String bugs.
xss            This plugin tests for XSS.
htaccessMethods This plugin searches for misconfigurations in the "<LIMIT>" configuration of Apache.
ssi            This plugin tests for server side inclusion bugs.
localFileInclude This plugin tests for local file inclusion bugs.
fileUpload     This plugin greps every page for forms with file upload capabilities.
```



# Audit Demo

## Lets perform an audit

- Scenario:
  - We do not know the site structure
  - we need to test the site for common xss and osCommanding vulnerabilities.

# Exploit : a closer look

- Exploit plugins [ab]use the vulnerabilities found in the audit phase and return something useful to the user ( remote shell, SQL table dump, a proxy, etc ).

```
w3af>>> exploit
w3af/exploit>>> list exploits
googleProxy      This plugin is a local proxy for HTTP requests that
davShell         This plugin exploits apps that have unauthenticated dav access.
rfiProxy         This plugin exploits remote file inclusions to create a proxy server.
osCommandingShell This plugin exploits OS Commanding bugs.
sqlmap           This plugin exploits [blind] sql injections using sqlmap ( http://sqlmap.sf.net ).
mysqlWebShell    This plugin exploits [blind] sql injections using to create a webshell on the remote host.Plugin author:
localFileReader  This plugin exploits local file inclusion bugs.
fileUploadShell  This plugin exploits apps that allow php's and asp's uploads to webroot.
xssBeef          This plugin exploits XSS vulnerabilities using beEF ( www.bindshell.net/tools/beef/ ) .
remoteFileIncludeShell This plugin exploits remote file include bugs.
w3af/exploit>>> █
```



# Exploit Demo 1

Lets exploit a OS Commanding  
vulnerability

- Scenario:
  - We have located an injection point and would like to test it for a osCommanding vulnerability.

# Exploit Demo 2

Lets exploit a sql injection  
vulnerability

- Scenario:
  - We have located an injection point that we would like to test for blind sqli.
  - parameter = id
  - parameter type integer

# Future

- Bug fixing and creating a robust core
- The web interface is actively being enhanced
- The current to-do list can be found at:

[http://sourceforge.net/pm/?group\\_id=170274](http://sourceforge.net/pm/?group_id=170274)



# How to get it

- Available through its main site  
<http://w3af.sourceforge.net>
- User guide essential reading
- Latest version can be obtained from:  
<https://w3af.svn.sourceforge.net/svnroot/w3af/>

# Conclusions

- The framework provides an architecture that allows growth.
- Excels at the discovery phase
- w3af detects / audits and exploits a large number of vulnerabilities in contrast to some other tools.
- The project needs source contributions.

*“complete knowledge about the framework and it's usage is complex and can be achieved only by using it.”*

# Questions

